EURASIP Journal on Advances in Signal Processing

RESEARCH Open Access

Object tracking method based on edge detection and morphology

Jie Xu[1], Sijie Niu[1*] and Zhifeng Wang[1]

*Correspondence:
20221513061@stu.sspu.edu.cn

[1] College of Intelligent Manufacturing and Control Engineering, Shanghai Second Polytechnic University, Shanghai 201209, China

Abstract

With the continuous development of science and technology, intelligent surveillance technology using image processing and computer vision is also progressing. To improve the performance of target detection and tracking, an improved target tracking method is proposed, which uses a combination of the Canny operator and morphology for the detection part, and a Kalman filter extended Kernel Correlation Filter (KCF) tracking algorithm approach for the tracking part. First, a convolution kernel of 3 × 3 is improved to a convolution kernel of 2 × 2 in the traditional Canny algorithm, and the pixel gradient in the diagonal direction is increased. Secondly, a mathematical morphology theory of nonlinear filtering is applied to the Canny edge detection algorithm, and this method effectively improves the clarity of image edges. Finally, the extended kernel correlation filtering algorithm is applied to video surveillance and Online Object Tracking Benckmark2013 (OTB2013) datasets for testing. The experimental results show that the method proposed in this paper can accurately detect moving targets and the algorithm has good accuracy and success rate.

Keywords: OTB, Canny, Adaptive, Video surveillance, Single-object tracking

1 Introduction

In recent years, with the rapid development of robot autonomous navigation, automatic driving, intelligent monitoring and other technologies in the field of security [1–3], video and image-based personnel target tracking technology has become an important research area [4–7]. The camera is an important vision tool widely used in the field of machine vision, and the relevant information it obtains is the basis for the next decision [8–10]. Therefore, how to accurately detect moving objects and effective tracking are a research hotspot in the field of robot vision [11, 12]. Accurate detection of targets is the key to the success of the whole research. In the field of target detection, edge detection is usually used as a preprocessing step to help algorithms recognize and localize targets more accurately. At the same time, edge detection can also be used in image.

Segmentation and image enhancement [13–15], it is studied to identify the locations in an image with the most drastic grayscale changes, which usually manifests itself in identifying the edges or contours of objects in an image. Edge detection needs to be highly accurate, stable, and real-time, and can effectively resist the interference of noise, so it is necessary to choose the appropriate edge detection algorithms and techniques according to the specific scene and needs. Traditional edge detection operators such as the Sobel operator are not able to accurately extract image edges, although they can smooth the noise and eliminate it.

The canny operator is a multistage optimization operator that filters noise and detects the edges of an image. Before processing the image, the canny operator smooths the image using a Gaussian filter to remove noise. The operator statistically extracts the edge information of the image while extracting the magnitude and direction of the image gradient by finite difference of the first-order partial derivatives. During processing, the Canny operator will also perform the process of non-polarization suppression, which means keeping the gradient direction locally polarized to reduce noise and blurred regions in the edge detection results. Finally, the Canny operator uses two thresholds to connect the edges. However, the conventional Canny algorithm uses a 2 × 2 convolutional Kernel with only horizontal and vertical directions in the detection direction, which is not complete enough in the extraction of edge information, so this paper adopts a 3 × 3 convolutional kernel with one more gradient component selected in the diagonal direction.

Morphology theory is a combination of lattice theory and topology, which is a nonlinear filtering technique widely used in the field of image processing, as well as a filtering technique in the study of image edge detection and image segmentation. Therefore, edge detection using mathematical morphology can reduce the effect of noise and preserve the original image content with the best results.

In motion target tracking, the commonly used methods are Camshift-based algorithms [16–19], Algorithms based on Kalman filter [20, 1] as well as based on particle filtering [21–24]. The Camshift algorithm is an improved algorithm derived from the MeanShift algorithm. It was first described by Gary R. Bradski et al. It was proposed and applied to face tracking. However, the algorithm only uses color information for tracking and is prone to tracking errors when the background color is similar to the target. Particle filtering algorithm is an algorithm that can effectively deal with the problem of target tracking in complex environments; the algorithm approximates the a posteriori probability density of the system by utilizing a certain number of particles to achieve target tracking, but the algorithm has a high complexity due to the need for a large number of samples. The Kalman filter algorithm can track the target or assist the tracking process by constantly updating the state of the target, which effectively improves the quality of target tracking based on appearance features, reduces the object boundary tracking error, and narrows the scope of the candidate tracking region.

Therefore, to improve the effective detection and tracking of moving targets such as people, this paper improves the traditional Canny operator in the calculation of gradient magnitude and fuses it with the morphological edge detection algorithm, to accurately recognize the edge of the target and then get the target's position, based on which the improved kernel correlation filtering algorithm is introduced for effective tracking of pedestrians [25, 19, 26, 27]. The whole process of the algorithm is shown in Fig. 1.

2 Target detection algorithms

2.1 Canny edge detection principle

Canny edge detection is a multi-step algorithm and has gradually improved the evaluation criteria in theory, the main steps of this edge detection algorithm include Gaussian filtering, calculation of gradient magnitude and direction, non-maximum value

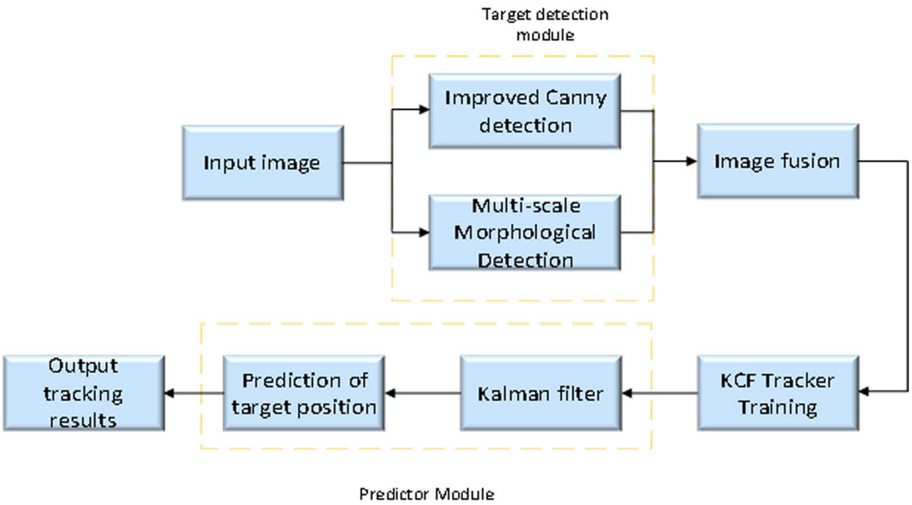


Fig. 1 Process diagram of the target tracking algorithm including detection and prediction modules

suppression, and double threshold processing. Among them, Gaussian filtering can effectively reduce noise, gradient magnitude and direction can extract the edge information in the image, non-extremely large value suppression can retain the detailed information of the edge, and double thresholding can distinguish the edge from the noise. The advantage of the Canny edge detection operator is that it can detect various types of edges, including strong edges and weak edges, and it can accurately locate the position of the edges. Therefore, the Canny edge detection operator is widely used in image processing and analysis in the field of computer vision. It follows the following steps in detecting the image edges.

1. Smooth the image and remove noise using Gaussian filtering.
2. Derive the Gaussian filter to obtain the magnitude and direction of the gradient along the x- and y-dimensions.
3. Non-maximum suppression of the gradient magnitude, retaining only the local maxima of the gradient.
4. Detecting and connecting edges with a double-thresholding algorithm.

This edge detection operator has good localization performance and can effectively suppress the low probability multiple responses of single edges and false edges, so it can extract the edge information of complex background images. The steps of the algorithm are shown in Fig. 2.

2.2 Improved Canny edge detection operator

Since the traditional Canny edge detection operator uses a 2 × 2 convolutional kernel, unable to extract effective edge information, therefore, in this paper, while retaining the convolutional template of the Sobel operator, we improve the differential template near 2 × 2, change the convolution kernel to 3 × 3 and add the differential template of the first-order partial derivatives in the 45° and 135° directions, the template representation of the eight-neighborhood of the center pixel in the gradient template is given as

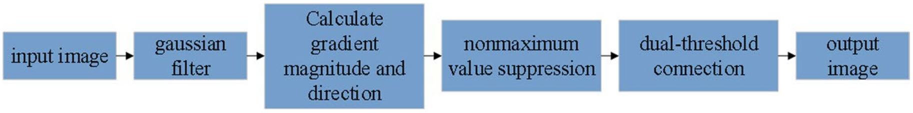

input image
gaussian filter
Calculate gradient magnitude and direction
nonmaximum value suppression
dual-threshold connection
output image

Fig. 2 Traditional Canny algorithm flow

$$S_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} S_r = \begin{pmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{pmatrix}$$
$$S_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} S_l = \begin{pmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{pmatrix} \tag{1}$$

Here, the convolution kernels for the x direction, y direction, 45° direction, and 135° direction are denoted as S_x, S_y, S_r, S_l, respectively, and the operations are performed using the convolution kernels while processing the image so that the gradient components g_x, g_r, g_y and g_l can be obtained in each of the different directions. Finally, the gradient magnitude and gradient direction angle of the image are

$$G = \sqrt{g_x^2 + g_y^2 + g_r^2 + g_l^2} \tag{2}$$

$$\theta = \arctan\left(\frac{g_x}{g_y}\right) \tag{3}$$

2.3 Improved Canny edge detection with morphological fusion

Multiscale morphological edge detection methods mainly utilize structural features at different scales to detect edges in images. This method can capture edges of different sizes and shapes at different scales. Specifically, this method utilizes a set of structural units of varying sizes to corrode and swell the image, based on which the edge image can be obtained by differentiating the above-processed image. The advantage of this method is that it is able to remove noise and details, which improves the detection accuracy of the edges. Therefore, in order to get a better filtering effect on the image, this paper fuses the multiscale morphology and the improved Canny edge detection operator.

To increase the noise immunity of the improved algorithm, the image can be detected using three structural elements of different sizes. This has the advantage of suppressing noise and getting a more complete edge profile. The structuring elements chosen for this paper are as follows

$$\begin{aligned} B_1 &= \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad B_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \\ B_3 &= \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{aligned} \tag{4}$$

Firstly, the outermost edge of the image is extracted using the noise-resistant expansion operator and then based on this, the inner boundary of the image is extracted using this denoising operator. An edge detection formula for resisting noise is

$$y_d = \left(\left(\left(f \circ B_1\right) \cdot B_2 \cdot B_3\right) \circ B_3 \oplus B_1\right) \circ B_1 - \left(\left(f \circ B_1\right) \cdot B_2 \cdot B_3\right) \circ B_3 \tag{5}$$

$$y_e = \left(\left(f \circ B_1\right) \cdot B_2 \cdot B_3\right) \circ B_3 - \left(\left(\left(f \circ B_1\right) \cdot B_2 \cdot B_3\right) \Theta B_1\right) \cdot B_1 \tag{6}$$

In the formula, f is the input image, y_d represents the outer edge contour of the image, y_e represents the inner edge contour of the image, the expansion operation is $\oplus$, the erosion operation is Θ, $\circ$ is the open operation, and $\cdot$ is the closed operation.

Finally, the obtained inner and outer edges, as well as the edges bounded between the inner and outer edges, are weighted and fused to obtain an enhanced noise-resistant morphological edge detection algorithm.

$$E_{\max} = \max\left(y_e, y_d\right) \tag{7}$$

$$E_{\min} = \min\left(y_e, y_d\right) \tag{8}$$

$$E_a = y_d + y_e + 1.2(E_{\max} - E_{\min}) \tag{9}$$

In order to obtain a well-defined image, this paper fuses multiscale morphology with the improved Canny detection operator, and the specific flowchart is shown in Fig. 3.

In the figure, the input image is a color image and the output image is the resultant image of edge detection by fusing Canny edge detection operator and multiscale morphology; the two algorithms are weighted and fused through the double threshold connection described in the literature, the steps are as follows:

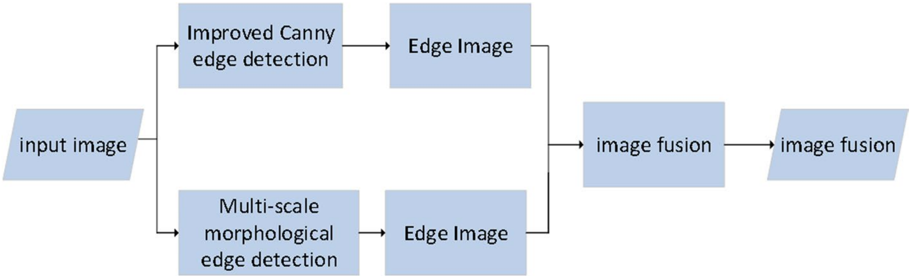

input image
Improved Canny edge detection
Edge Image
Multi-scale morphological edge detection
Edge Image
image fusion
image fusion

Fig. 3 Algorithmic process

1. First, the image is edge detected using the improved Canny operator, and the morphological edge detection operator to obtain the edge images B_1 and B_2;

2. Then, the two images are subtracted to obtain the dissimilarity map between B_1 and B_2, B_1 and B_d;

$$B_d = B_1 - B_2 \tag{10}$$

3. Find pixel points in the difference map that have a gray value other than 0, and then mark the edge points adjacent to them as 1 in the neighborhood;

4. Calculate the number of 1's around the center pixel point, if the number of 1 's is four, then, this point is defined as a weak edge point of the image, and finally, the detail edge map b is obtained;

5. Fuse the edge image B_1 and B_2 with the edge map b to get a clearer edge map.

$$B = B_1 * B_2 + 1.5b \tag{11}$$

3 Improved kernel correlation filter tracking algorithm based on Kalman filtering

Firstly, we briefly introduce the process of target detection, the main steps of detecting moving objects are difference operation, binarization processing, morphological processing, edge extraction, and finally obtaining the capture of moving objects. Since morphological processing and edge extraction have been introduced in the previous part of the article, we will not repeat them here. The concepts of difference operation and binarization are introduced below.

1. Differential Operations

The difference operation of the image, which is mainly to do the difference operation between the pixels of the current frame and the background pixels, so as to initially obtain the moving object, its mathematical formula is shown as follows?

$$Diff(x,y,frame) = I(x,y,frame) - B(x,y) \tag{12}$$

2. Binarization

By setting a threshold, the foreground obtained by subtracting the background is a binary image.

$$F(x,y,frame) = \begin{cases} 0, & Diff(x,y,frame) \le T \\ 1, & Diff(x,y,frame) > T \end{cases} \tag{13}$$

Diff(x, y, frame) denotes the gray value at the pixel position (x, y) under the frame, T denotes the segmentation threshold, and F(x, y, frame) denotes the binary image. On this basis, the threshold T is set to segment the grayscale image and distinguish the image from the background. Selecting too high a threshold value will not be able to completely segment the moving objects, and selecting too low a threshold value will generate a lot of noise. Therefore, the threshold needs to be correctly selected to minimize the background interference and noise. Finally, the detection results are input into the tracker.

3.1 Moving object tracking

The traditional kernel correlation filtering algorithm is a discriminative target tracking algorithm [28–30]; the core of the KCF algorithm is to collect positive and negative samples in the target area through the cyclic matrix and train the target using the ridge regression algorithm and then, transform the matrix into the dot product of elements through a diagonalizable property in the cyclic matrix in the Fourier space, which reduces the computational cost, improves the tracking speed, and enables the algorithm to satisfy the real-time requirements. However, the traditional KCF algorithm often needs to manually box the target tracking box; the algorithm accuracy is not high, and when the target occurs occlusion and overlap easily lead to tracking failure, while the Kalman filter algorithm is an algorithm to track the target by constantly updating the target state [31–33], which can be used for target tracking or assisting the tracking process. Kalman filtering realizes target tracking by fusing the prediction of the target state and the measured values, which can effectively improve the quality of target tracking based on appearance features, reduce the object boundary tracking error, and narrow the range of candidate tracking areas.

3.1.1 Kalman filter principle

Kalman filtering is an algorithm used for state estimation, the basic principle of which is to realize state estimation by fusing predictions and observations of the system state. In target tracking, Kalman filtering can be used to estimate state quantities such as position, velocity and acceleration of the target. The state transfer equation for Kalman filtering is:

$$x_{k+1} = A_k x_k + \omega_k \tag{14}$$

The principle of this equation is to estimate the current state based on the state and control variables of the previous moment, ω_k is a noise obeying a Gaussian distribution, and A_k is a state transfer matrix. The target observation equation is:

$$z_k = H_k x_k + v_k \tag{15}$$

v_k is the observed noise, obeying a Gaussian distribution, also known as measurement noise. The state update phase of Kalman filtering is based on the predicted state values and the observed residuals to compute the state estimates and the state covariance matrix.

Kalman filter gain coefficient calculation:

$$K_k = P_k^- H^T \left(H P_k^- H^T + R_k \right)^{-1} \tag{16}$$

State variable update:

$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - H_k \hat{x}_k^- \right) \tag{17}$$

Error covariance values are updated:

$$P_k = (I - K_k H_k) P_k^- \tag{18}$$

3.1.2 Principle of kernel correlation filtering

By using a one-dimensional vector $x = (x_1, x_2, \ldots, x_n)$ as the base sample, the sample is sampled using the concept of a cyclic matrix:

$$X = C(x) = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix} \tag{19}$$

The ridge regression expression for this algorithm is?

$$f(x_i) = w^T \varphi(x_i) \tag{20}$$

Putting w in terms of a linear combination of samples as:

$$w = \sum_i \alpha_i \varphi(x_i) \tag{21}$$

For the regression function, the purpose of this training is to use x_i to minimize the error of the regression objective y_i, the above can be attributed to training the classifier. According to the nature of the cyclic matrix, the Fourier transform of Eq can simplify the equation as:

$$\hat{f}(z) = \hat{k}^{xz} \odot \hat{\alpha} \tag{22}$$

Here, $\odot$ means dot product, $^\wedge$ stands for Fourier variation; x is the base sample, z is the training sample, k^{zx} is the kernel relationship between the base sample and the training sample; α stands for vector coefficients.

3.1.3 Kalman filter-based kernel correlation filter tracking

This paper proposes a kernel correlation filter tracking algorithm extended by Kalman filter. Firstly, the moving target is detected by the improved target edge detection algorithm in the previous paper, and the KCF tracking frame is initialized at the same time; then, once the target is occluded or lost, the position of the target is predicted by using the Kalman filter, and this tracking algorithm is able to achieve stable tracking of the target. The steps of the improved tracking algorithm are shown below.

1. Read video image sequences;
2. The specific position of the moving target is extracted by the target edge detection algorithm and marked with a rectangular box;
3. Calculate the initial position of the target according to the KCF algorithm;
4. Initialize the Kalman filter and predict the position of the target in the next frame.

4 Experimental results and analysis

4.1 Experimental environment

In order to verify the superiority of the improved algorithm in this paper, the relevant platform configuration for this simulation experiment is as follows: hardware environment CPU Intel(R) Core(TM)-i7-4200H@3.4GHz, The RAM is 16GB, running on Windows 11 operating system with Visual Studio 2019+MATLAB2021b running software.

4.2 Evaluation indicators

4.2.1 Evaluation indicators for target detection

In order to confirm the effectiveness of the edge detection algorithm proposed in the previous section, the edge detection effect of the image is evaluated by calculating the root mean square error (MSE) [34–37] of the image processed by the edge detection algorithm and the peak signal-to-noise ratio (PSNR) [38] of the image. The specific calculation methods are as follows

$$\mathrm{MSN} = \frac{1}{\mathrm{MN}} \sum_{\mathrm{i=1}}^{\mathrm{M}} \sum_{\mathrm{j-1}}^{\mathrm{N}} [\mathrm{I(i,j)} - \mathrm{K(i,j)}]^2 \tag{23}$$

$I(i, j)$ is the pixel value of the original image at (i, j), $K(i, j)$ is the pixel value of the processed image at (i, j), M and N are the number of pixels in the original and processed images, respectively. MSE is used to measure the difference between the processed image and the original image, when the value of MSN is smaller, the better the effect of image detection.

The Peak Signal to Noise Ratio of an image, i.e., PSNR is used to measure the similarity between the processed image and the original image and is calculated as follows:

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{\mathrm{MAX}_{\mathrm{I}}^2}{\mathrm{MSN}} \right) \tag{24}$$

$\mathrm{MAX}_{\mathrm{I}}$ is the maximum value of the pixel value, usually 255.The edge detection effect of the image can be evaluated by calculating the MSE and PSNR of the image processed by different algorithms. The smaller the MSE and the larger the PSNR, the closer the processed image is to the original image and the better the processing effect. Table 1 gives the comparison results of MSE and PNSR of the edge image after several different edge detection algorithms process the noise-free image. From the data in the table, it can be seen that the edge detection algorithm in this paper can better preserve the detail information of the image, the PSNR index is higher than the rest of the algorithms, and the MSE index is lower than the rest of the algorithms, which indicates that the image is smoother after processing by the algorithm in this paper, and it has better results.

Table 1 Results after processing by different edge detection algorithms

Evaluation indicators	Canny calculator	Prewitt calculator	Multiscale morphology	The detection algorithm in this paper
PSNR	58.3444	57.994	58.9087	59.386
MSE	0.0952	0.1032	0.0836	0.0749

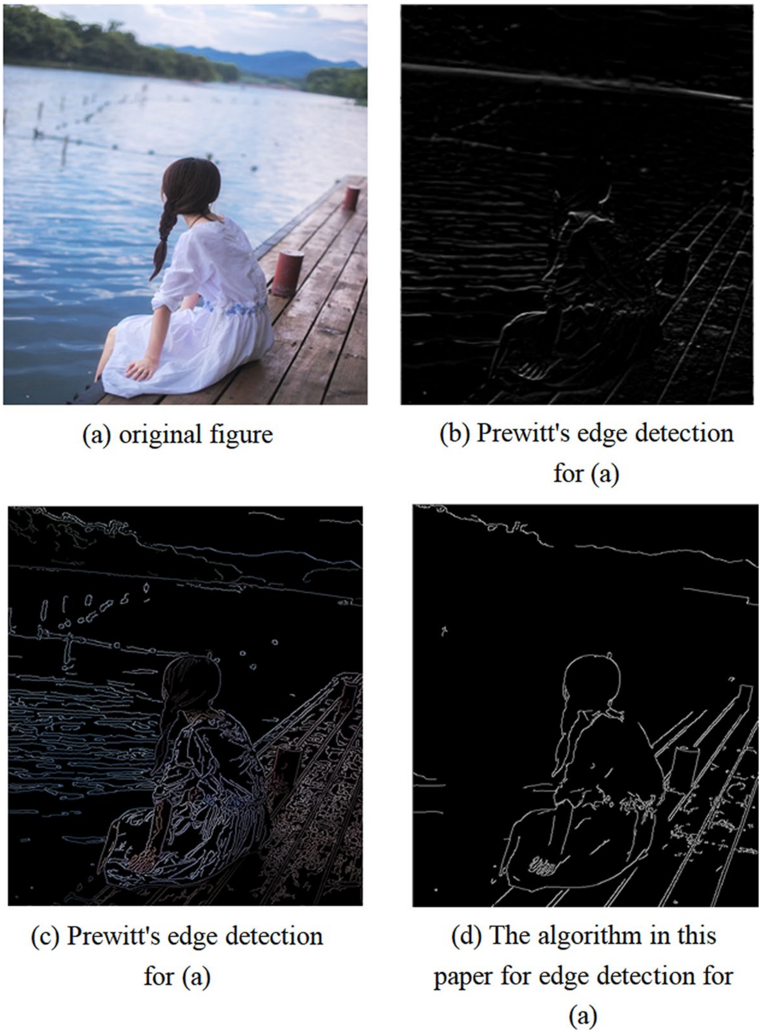
4.2.2 Targeted tracking evaluation indicators

The metrics for evaluating this paper's algorithm as well as other tracking algorithms in this experiment include the average distance precision (DP) [39] and the average overlap precision (OP) [40]. Distance precision is the percentage of frames with center error below a certain threshold (usually 20) to the total number of video frames, which can effectively reflect the robustness of the algorithm. Success rate indicates the percentage of frames with overlap greater than a threshold to the total number of video frames.

4.3 Detection tracking performance analysis

4.3.1 Qualitative inorganic analysis

In this paper, a color image is first used as a test image and tested and compared with the edge detection algorithm, and the comparison results are shown in Fig. 4. The figure shows a color map and the edge image processed by the three algorithms; from the figure, it can be seen that although all four algorithms can extract the edge information, but the algorithm proposed in this paper is clearer and more accurate


(a) original figure

(b) Prewitt's edge detection for (a)

(c) Prewitt's edge detection for (a)

(d) The algorithm in this paper for edge detection for (a)

Fig. 4 Detection results of different edge detection algorithms for portraits

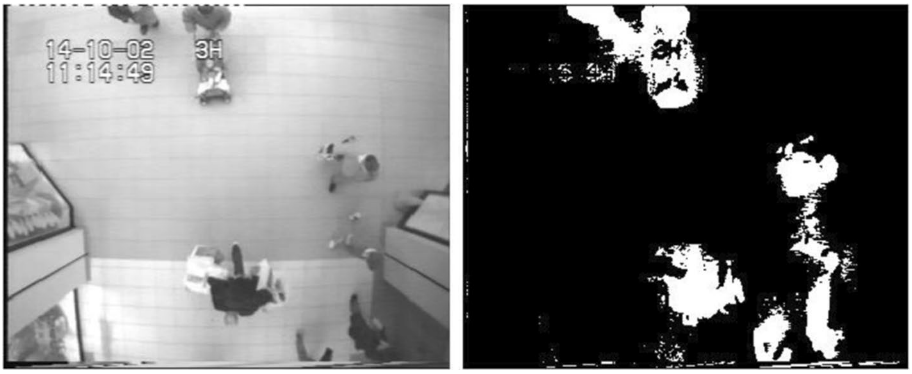


Fig. 5 Original video and differential binarization processed video

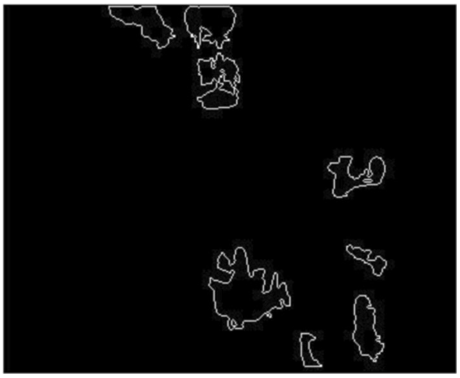


Fig. 6 Edge extraction simulation effect

compared to Fig. (b) and Fig. (c) contour, better completeness and smoothness, and the algorithm in this paper therefore has a better practicality.

In this paper, a section of the surveillance video is randomly selected as a test video, the current frame and the video background to do the difference operation, to obtain the image and then do the binarization process; the simulation results obtained are shown in Fig. 5.

From the simulation results of the original video and the video after differential binarization in Fig. 4, it can be seen that there is a large amount of noise interference in the figure after differential binarization, and most of the noise as well as the motion interference can be removed by morphological processing, which improves the processing effect of the whole system.

The simulation effect of edge extraction is shown in Fig. 6.

Through the above steps, the detection of the moving human body is finally realized and the detection results are shown in Fig. 7.

From the simulation results above, it can be seen that after the algorithm, it basically realizes the detection of each moving individual in the video; in the figure, the red part is the actual moving human body. Then, the tracking algorithm of this paper is applied to track the video sequence soccer in the standard test set OTB2013, as

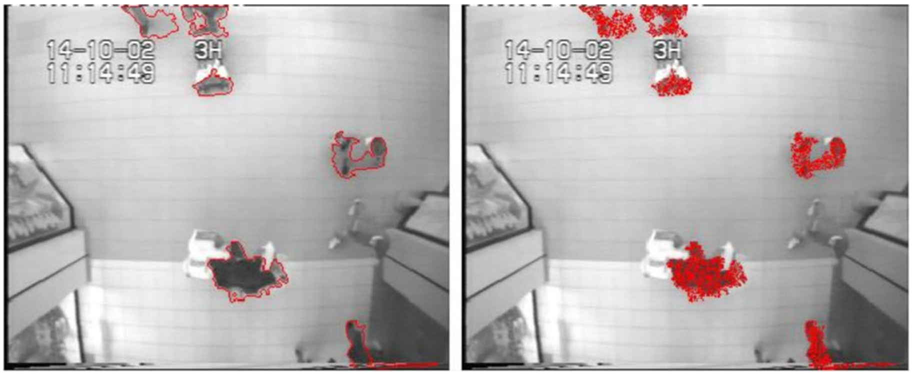


Fig. 7 Moving figure detection

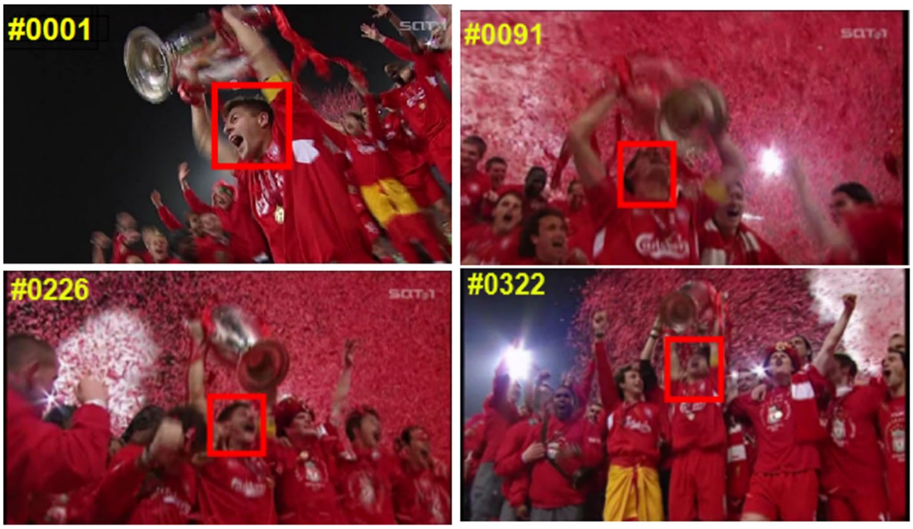


Fig. 8 Motion body tracking simulation effect

shown in Fig. 8. From the above simulation results, it is clear that a particular moving figure can be successfully tracked using this algorithm.

The following Fig. 9 shows the tracking failures of other trackers, the red target box is tracked using this algorithm, the yellow target box is tracked using the Discriminative Scale Space Tracker (DSST) algorithm, and the blue target box is tracked using the original KCF algorithm; from the figures, it can be seen that the tracker in this paper is able to accurately track the characters, and the other trackers have experienced varying degrees of tracking frame drifting.

4.3.2 Quantitative analysis

In this paper, several more typical algorithms are selected for comparison; they are the original KCF algorithm, Spatially Regularized Correlation Filters (SRDCF) algorithm, Sum of Template And Pixel-wise LEarners (STAPLE) algorithm, and Discriminative Scale Space Tracker (DSST) algorithm, while the algorithms are tested on the OTB2013 dataset. Fig. 10 shows the accuracy and success graphs obtained for the dataset video sequences. Through the experimental results, it can be seen that the accuracy of the

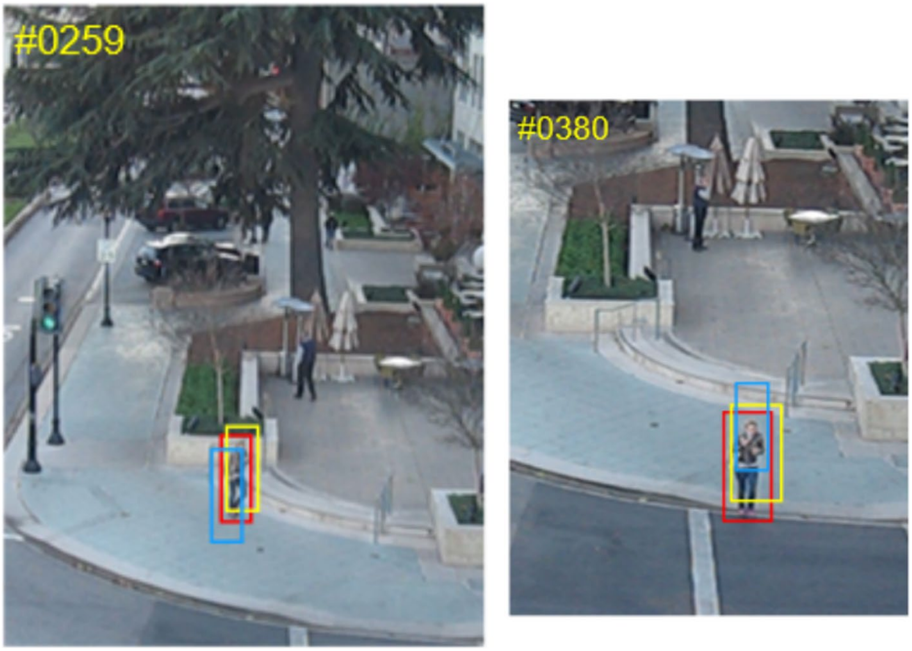


Fig. 9 Tracking effect of different trackers

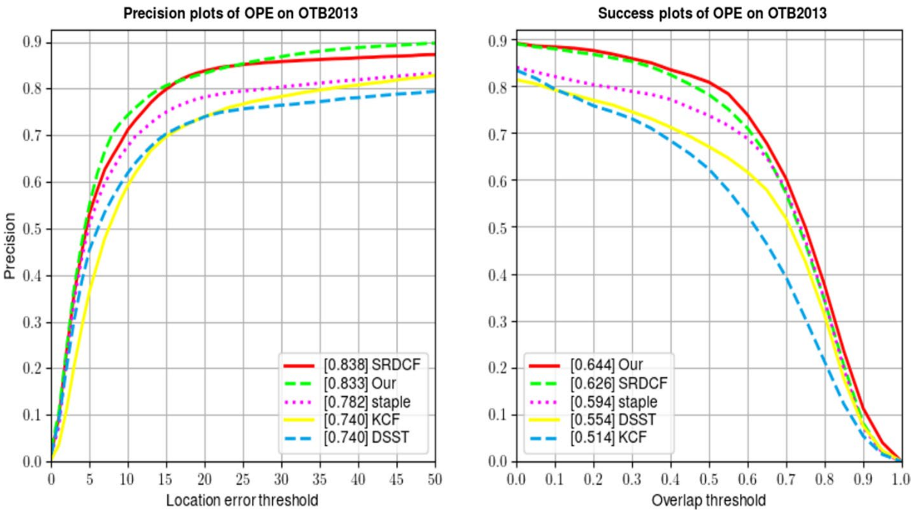


Fig. 10 Plot of success and accuracy rates of different algorithms

improved algorithm Ours in this paper is 83.3%, which is slightly better than the previous algorithm, and the success rate is 64.4%, which is also better than the pre-improved algorithm.

4.3.3 Ablation experiment

In order to check that the image edge detection model and the target position prediction model added in this paper are improving the performance of the tracker, the KCF tracker model is used as the baseline, and the ablation experiments are performed on

Table 2 Ablation studies

Method	Detection module	Predictor module	DP (%)	OP (%)
baseline	–	–	74.0	51.4
baseline + column 1	✓	–	78.6	56.3
baseline + column 2	–	✓	80.6	63.1
baseline + column 1–2	✓	✓	83.3	64.4

these two components, respectively; the results are shown in Table 2, when the target detection module is added, the DP(%) is improved, and the OP(%) is improved. The target tracking algorithm in this paper incorporates the above two models, and the tracking performance is improved compared to the original KCF tracking algorithm.

5 Conclusion

The paper proposes a target detection and tracking method applicable to intelligent surveillance scenarios. This method can significantly improve the accuracy and success rate of target tracking in intelligent surveillance scenarios. By improving the Canny edge detection operator while integrating the relevant algorithms of mathematical morphology, the clarity of image edge detection is effectively improved, and finally the extended kernel correlation filtering algorithm is applied to video surveillance and tracking in the standard test set OTB2013. The experimental results show that the method can accurately detect moving targets in the field of video surveillance and perform real-time tracking.

Abbreviations

KCF	Kernel Correlation Filter
OTB2013	Online object tracking benckmark2013
MSE	Mean square error
PSNR	Peak signal-to-noise ratio
DP	Distance precision
OP	Overlap precision
SRDCF	Spatially Regularized Correlation Filters
STAPLE	Sum of Template And Pixel-wise Learners
DSST	Discriminative Scale Space Tracker

Acknowledgements
The authors would like to thank the anonymous reviewers for their valuable comments and suggestions that helped improve the quality of the manuscript.

Author contributions
All authors take part in the discussion of the work described in this paper. These authors contributed equally to this work.

Funding
Not applicable.

Availability of data and materials
The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare that they have no Conflict of interest.

Received: 6 January 2024 Accepted: 22 March 2024
Published online: 03 April 2024

References

1. P. Dendorfer, A. Osep, A. Milan, K. Schindler, D. Cremers, I. Reid, S. Roth, L. Leal-Taixé, Motchallenge: a benchmark for single-camera multiple target tracking. Int. J. Comput. Vis. **129**, 845–881 (2021)
2. S. Shinde, A. Kothari, V. Gupta, Yolo based human action recognition and localization. Procedia Comput. Sci. **133**, 831–838 (2018)
3. O. Barnich, M. Van Droogenbroeck, Vibe: a universal background subtraction algorithm for video sequences. IEEE Trans. Image Process. **20**(6), 1709–1724 (2010)
4. N.-D. Hoang, Q.-L. Nguyen, A novel method for asphalt pavement crack classification based on image processing and machine learning. Eng. Comput. **35**, 487–498 (2019)
5. H. Li, Y. Li, F. Porikli, Deeptrack: learning discriminative feature representations online for robust visual tracking. IEEE Trans. Image Process. **25**(4), 1834–1848 (2015)
6. J. Sun, E. Ding, D. Li, A. Akram, M.K. Kerns, Long-term object tracking based on improved continuously adaptive mean shift algorithm. J. Eng. Sci. Technol. Rev. **13**(5), 33–41 (2020)
7. J. Wang, W. Liu, W. Xing, S. Zhang, Visual object tracking with multi-scale superpixels and color-feature guided kernelized correlation filters. Signal Process. Image Commun. **63**, 44–62 (2018)
8. Z. Sun, Y. Wang, R. Laganiere, Hard negative mining for correlation filters in visual tracking. Mach. Vis. Appl. **30**(3), 487–506 (2019)
9. M. Zolfaghari, H. Ghanei-Yakhdan, M. Yazdi, Real-time object tracking based on an adaptive transition model and extended Kalman filter to handle full occlusion. Vis. Comput. **36**, 701–715 (2020)
10. M.D. Jenkins, P. Barrie, T. Buggy, G. Morison, Extended fast compressive tracking with weighted multi-frame template matching for fast motion tracking. Pattern Recogn. Lett. **69**, 82–87 (2016)
11. J.F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2014)
12. W. Wei, J. Wang, Z. Fang, J. Chen, Y. Ren, Y. Dong, 3u:jJoint design of UAV–USV–UUV networks for cooperative target hunting. IEEE Trans. Veh. Technol. **72**(3), 4085–4090 (2022)
13. J. Yun, D. Jiang, Y. Liu, Y. Sun, B. Tao, J. Kong, J. Tian, X. Tong, M. Xu, Z. Fang, Real-time target detection method based on lightweight convolutional neural network. Front. Bioeng. Biotechnol. **10**, 861286 (2022)
14. F. Lei, F. Tang, S. Li, Underwater target detection algorithm based on improved yolov5. J. Mar. Sci. Eng. **10**(3), 310 (2022)
15. D. Bai, Y. Sun, B. Tao, X. Tong, M. Xu, G. Jiang, B. Chen, Y. Cao, N. Sun, Z. Li, Improved single shot multibox detector target detection method based on deep feature fusion. Concurr. Comput. Pract. Exp. **34**(4), 6614 (2022)
16. M. Zhao, W. Li, L. Li, J. Hu, P. Ma, R. Tao, Single-frame infrared small-target detection: a survey. IEEE Geosci. Remote Sens. Mag. **10**(2), 87–119 (2022)
17. S. Khan, M. Tufail, M.T. Khan, Z.A. Khan, J. Iqbal, A. Wasim, A novel framework for multiple ground target detection, recognition and inspection in precision agriculture applications using a UAV. Unmanned Syst. **10**(01), 45–56 (2022)
18. J. Zhang, Y. Liu, H. Liu, J. Wang, Y. Zhang, Distractor-aware visual tracking using hierarchical correlation filters adaptive selection. Appl. Intell. **52**(6), 6129–6147 (2022)
19. C. Liang, Z. Zhang, X. Zhou, B. Li, S. Zhu, W. Hu, Rethinking the competition between detection and reid in multiobject tracking. IEEE Trans. Image Process. **31**, 3182–3196 (2022)
20. P. Uijtewaal, P.T. Borman, P.L. Woodhead, C. Kontaxis, S.L. Hackett, J. Verhoeff, B.W. Raaymakers, M.F. Fast, First experimental demonstration of VMAT combined with MLC tracking for single and multi fraction lung SBRT on an MR-linac. Radiother. Oncol. **174**, 149–157 (2022)
21. S. Liu, X. Liu, S. Wang, K. Muhammad, Fuzzy-aided solution for out-of-view challenge in visual tracking under IoT-assisted complex environment. Neural Comput. Appl. **33**, 1055–1065 (2021)
22. B. Li, Y. Wu, Path planning for UAV ground target tracking via deep reinforcement learning. IEEE access **8**, 29064–29074 (2020)
23. X. Yang, G. Huang, F. Li, X. Li, B. Li, C. Geng, X. Li, Continuous tracking and pointing of coherent beam combining system via target-in-the-loop concept. IEEE Photonics Technol. Lett. **33**(20), 1119–1122 (2021)
24. M. Li, Z. Cai, J. Zhao, Y. Wang, Y. Wang, K. Lu, MNNMs integrated control for UAV autonomous tracking randomly moving target based on learning method. Sensors **21**(21), 7307 (2021)
25. A. Dak, R. Radhakrishnan, Tracking and interception of a spiralling ballistic target on reentry. IFAC-PapersOnLine **55**(1), 339–344 (2022)
26. J. Liu, Z. Wang, M. Xu, DeepMTT: a deep learning maneuvering target-tracking algorithm based on bidirectional LSTM network. Inf. Fusion **53**, 289–304 (2020)
27. S. Liu, S. Wang, X. Liu, C.-T. Lin, Z. Lv, Fuzzy detection aided real-time and robust visual tracking under complex environments. IEEE Trans. Fuzzy Syst. **29**(1), 90–102 (2020)
28. M. Martinez-Garcia, Y. Zhang, T. Gordon, Memory pattern identification for feedback tracking control in human–machine systems. Hum. Factors **63**(2), 210–226 (2021)
29. Q. Liu, D. Yuan, N. Fan, P. Gao, X. Li, Z. He, Learning dual-level deep representation for thermal infrared tracking. IEEE Trans. Multimed. **25**, 1269–1281 (2022)
30. Y. Sun, C. Wang, Z. Wang, J. Liao, Efficient algorithm for tracking the single target applied to optical-phased-array lidar. Appl. Opt. **60**(35), 10843–10848 (2021)
31. J. Chen, B. Huang, J. Li, Y. Wang, M. Ren, T. Xu, Learning spatio-temporal attention based siamese network for tracking UAVs in the wild. Remote Sens. **14**(8), 1797 (2022)

32. S.A. Memon, M.-S. Park, I. Memon, W.-G. Kim, S. Khan, Y. Shi, Modified smoothing algorithm for tracking multiple maneuvering targets in clutter. Sensors **22**(13), 4759 (2022)
33. P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, H. Ling, Detection and tracking meet drones challenge. IEEE Trans. Pattern Anal. Mach. Intell. **44**(11), 7380–7399 (2021)
34. S. Liu, S. Wang, X. Liu, A.H. Gandomi, M. Daneshmand, K. Muhammad, V.H.C. De Albuquerque, Human memory update strategy: a multi-layer template update mechanism for remote visual monitoring. IEEE Trans. Multimed. **23**, 2188–2198 (2021)
35. J. Zhang, J. Sun, J. Wang, Z. Li, X. Chen, An object tracking framework with recapture based on correlation filters and siamese networks. Comput. Electr. Eng. **98**, 107730 (2022)
36. Y. Wang, T. Wang, G. Zhang, Q. Cheng, J.-Q. Wu, Small target tracking in satellite videos using background compensation. IEEE Trans. Geosci. Remote Sens. **58**(10), 7010–7021 (2020)
37. W. Song, Z. Wang, J. Wang, F.E. Alsaadi, J. Shan, Distributed auxiliary particle filtering with diffusion strategy for target tracking: a dynamic event-triggered approach. IEEE Trans. Signal Process. **69**, 328–340 (2020)
38. Z. Li, J. Xie, W. Liu, H. Zhang, H. Xiang, Joint strategy of power and bandwidth allocation for multiple maneuvering target tracking in cognitive MIMO radar with collocated antennas. IEEE Trans. Veh. Technol. **72**(1), 190–204 (2022)
39. L. Guo, W. Liu, L. Li, Y. Lou, X. Wang, Z. Liu, Neural network non-singular terminal sliding mode control for target tracking of underactuated underwater robots with prescribed performance. J. Mar. Sci. Eng. **10**(2), 252 (2022)
40. G. Jin, Player target tracking and detection in football game video using edge computing and deep learning. J. Supercomput. **78**(7), 9475–9491 (2022)

Publisher's Note